

1 **CLAIMS**

2

3 1. A method for providing thread scheduling in a device, the device

4 comprising one or more hardware elements operatively coupled to an operating

5 system comprising a plurality of program modules, the method comprising:

6 scheduling one or more threads according to a predetermined periodic rate;

7 determining whether or not there are any threads to execute; and

8 responsive to a determination that there are no threads to execute,

9 deactivating at least one subset of components for a dynamic variable amount of

10 time, the one subset being selected from a group of components comprising the

11 hardware elements and the program modules, the dynamic variable amount of time

12 being independent of the predetermined periodic rate and being based on a sleep

13 state of a set of threads in a sleep queue.

14

15 2. A method as recited in claim 1, wherein the dynamic variable amount

16 of time is based on a maximum amount of time that a thread can yield before

17 needing to be scheduled for execution.

18

19 3. A method as recited in claim 1, wherein the device is a battery

20 powered device.

21

22

23

24

25

1 4. A method as recited in claim 1, wherein the operating system
2 comprises an operating system selected from a group of operating systems
3 comprising Microsoft WINDOWS CE, Linux, WindRiver, QNX, or PALM
4 operating systems.

5
6 5. A method as recited in claim 1, wherein the scheduling, the
7 predetermined periodic rate is a millisecond.

8
9 6. A method as recited in claim 1:
10 wherein the providing further comprises setting a system timer to generate
11 a notification at the predetermined periodic rate;

12 wherein the deactivating further comprises resetting the system timer to
13 generate the notification after the dynamic variable amount of time has elapsed
14 since the deactivating; and

15 wherein the method further comprises:
16 receiving the notification after the dynamic variable amount of time
17 has elapsed since the deactivating; and

18 responsive to the receiving:
19 resetting the system timer to generate the notification at the
20 predetermined periodic rate; and
21 activating the at least one subset of components.

22
23 7. One or more computer-readable media containing a computer
24 executable program that performs a method as recited in claim 1.
25

1 8. A method for providing thread scheduling in a device, the device
2 comprising one or more hardware elements operatively coupled to an operating
3 system comprising a plurality of program modules, the method comprising:

4
5 scheduling one or more threads at a predetermined periodic rate;

6 determining whether or not there are any threads to execute; and

7 responsive to a determination that there are no threads to execute,
8 deactivating at least one subset of components for a dynamic variable amount of
9 time, the one subset being selected from a group of components comprising the
10 hardware elements and the program modules, the dynamic variable amount of time
11 being based on a sleep state of the a set of threads in a sleep queue and
12 independent of the predetermined periodic rate; and

13 activating the one subset of components only when the operating system
14 needs to perform an action selected from a group of actions comprising scheduling
15 a thread for execution upon expiration of the dynamic variable amount of time
16 since the deactivating, or upon receipt of an external event, processing the external
17 event, wherein the external event is not a system timer event.

18
19 9. A method as recited in claim 8, wherein the device comprises a
20 battery powered device.

21
22 10. A method as recited in claim 8, wherein the operating system
23 comprises a Microsoft WINDOWS CE operating system.
24
25

1 **11.** A method as recited in claim 8, wherein the scheduling, the
2 predetermined periodic rate is a millisecond.

3
4 **12.** A method as recited in claim 8:
5 wherein the scheduling further comprises setting a system timer to the
6 predetermined periodic rate, the predetermined periodic rate corresponding to a
7 thread scheduling accuracy; and

8 wherein the deactivating further comprises resetting the system timer to
9 generate a notification after the dynamic variable amount of time has elapsed since
10 the deactivating.

11
12 **13.** A method as recited in claim 8:
13 wherein the deactivating further comprises resetting a system timer to
14 generate a notification after the dynamic variable amount of time has elapsed, the
15 dynamic variable amount of time being a maximum amount of time that a thread
16 can yield to other threads before needing to be scheduled for execution; and

17 wherein the activating further comprises resetting the system timer to the
18 predetermined periodic rate to provide substantial thread scheduling accuracy.

19
20 **14.** One or more computer-readable media containing a computer
21 executable program that performs a method as recited in claim 8.
22
23
24
25

1 **15.** A computer-readable storage medium containing computer-
2 executable instructions for scheduling threads in a device, the device including an
3 operating system comprised of a plurality of program modules that are in turn
4 coupled to one or more hardware elements, the computer-executable instructions
5 comprising instructions for:

6 determining at a periodic rate whether or not there are any threads to
7 execute; and

8 responsive to a determination that there are no threads to execute,
9 deactivating at least one subset of components for a dynamic variable amount of
10 time, the at least one subset being selected from a group of components
11 comprising the one or more of the program modules and one or more of the
12 hardware elements, the dynamic variable amount of time being independent of the
13 periodic rate, the dynamic variable amount of time being based on a sleep state of
14 a set of threads in a sleep queue.

15
16 **16.** A computer-readable storage medium as recited in claim 15,
17 wherein the dynamic variable amount of time comprises a maximum amount of
18 time that a thread has specified that it will yield to other threads before it needs to
19 be scheduled for execution.

20
21 **17.** A computer-readable storage medium as recited in claim 15,
22 wherein the device comprises a battery powered device.
23
24
25

1 **18.** A computer-readable storage medium as recited in claim 15,
2 wherein the operating system comprises a Microsoft WINDOWS CE operating
3 system.

4
5 **19.** A computer-readable storage medium as recited in claim 15,
6 wherein the computer-executable instructions further comprise instructions for:

7 in the deactivating, configuring a system timer to send a first timer interrupt
8 after the dynamic variable amount of time has elapsed, the dynamic variable
9 amount of time being a maximum amount of time that a first thread can yield to a
10 second thread before the first thread needs to be executed; and

11 responsive to receiving the first timer interrupt:

12 (a) configuring the system timer to send a second timer interrupt at
13 the periodic rate; and

14 (b) activating the deactivated at least one subset of components to
15 determine if there are any threads to execute.

16
17 **20.** A computer-readable storage medium as recited in claim 15,
18 wherein the computer-executable instructions further comprise instructions for:

19 receiving an external interrupt before the dynamic variable amount of time
20 has elapsed since the deactivating, the external interrupt not being a system timer
21 interrupt; and

22 responsive to receiving the external interrupt, processing the external
23 interrupt such that the at least one subset of components remain deactivated for the
24 dynamic variable amount of time.

1 **21.** A computer comprising one or more computer-readable media as
2 recited in claim 15.

3
4 **22.** A device comprising:
5 a processor configured to fetch and execute a plurality of computer-
6 executable instructions;
7 a plurality of hardware elements coupled to the processor; and,
8 a memory coupled to the processor for storing the computer-executable
9 instructions comprising a scheduler program module, a hardware abstraction layer
10 (HAL) program module, one or more operating system program modules, and a
11 set of application program modules;

12 wherein the scheduler comprises computer-executable instructions for:

13 scheduling threads for execution at a periodic time interval; and

14 determining that there are no threads to execute;

15 wherein the HAL, responsive to the determining, comprises computer-
16 executable instructions for deactivating, for a dynamic variable amount of time, at
17 least one subset of components selected from a group of components comprising
18 the scheduler, the hardware elements, the one or more operating system program
19 modules, and the application program modules, the dynamic variable amount of
20 time being independent of the periodic time interval and being based on a sleep
21 state of a set of threads in a sleep queue.

1 **23.** A device as recited in claim 22, wherein the dynamic variable
2 amount of time is based on a maximum amount of time that a thread can yield
3 before needing to be scheduled.
4

5 **24.** A device as recited in claim 22, wherein the periodic time interval is
6 a millisecond.
7

8 **25.** A device as recited in claim 22, wherein the device comprises a
9 battery powered device.
10

11 **26.** A device as recited in claim 22, wherein the operating system is an
12 operating system selected from a group of operating systems comprising Microsoft
13 WINDOWS CE, Linux, WindRiver, QNX, or Palm ® operating systems.
14

15 **27.** A device as recited in claim 22, wherein the HAL further comprises
16 computer-executable instructions for re-activating the at least one subset of
17 components after the dynamic variable amount of time has elapsed since the at
18 least one subset of components were deactivated.
19

20 **28.** A device as recited in claim 27, wherein the scheduler is re-activated
21 in a manner that allows the scheduler to schedule threads based on the periodic
22 time interval.
23
24
25

1 **29.** A device as recited in claim 22, wherein after the scheduler is
2 deactivated, the HAL further comprises computer-executable instructions for
3 receiving a notification in response to an external event, the external event not
4 being a system timer event, responsive to receipt of the notification, the HAL
5 processing the notification in a manner that the scheduler remains deactivated for
6 the dynamic variable amount of time.